Gini Index based Fuzzy classification system using Ant Colony Optimization for diabetes disease diagnosis

Muhammad Husnain Sardar
606-FBAS/MSCS/F10

Supervisor
Dr. Ayyaz Hussain
Assistant Professor
Department of Computer Science & Software Engineering,
International Islamic University Islamabad, Pakistan

Department of Computer Science & Software Engineering,
Faculty of Basic and Applied Sciences,
International Islamic University Islamabad, Pakistan
Acknowledgement

I want to express my sincere appreciativeness to my director, Assistant Professor Dr. Ayyaz Hussain, for his never ending supervision, sustenance and inspiration during the study of Master of computer science in International Islamic University, Islamabad, Pakistan at Department of Computer Science and Software Engineering. I am predominantly grateful to him for teaching me the art of research and thesis writing, which proved valuable to my current research work and for future career also. Without his sustenance, hard work, knowledge, and tolerance this research would be an impossible thing. The selection of experimental methods and the way of displaying the results presented in this thesis have been influenced by him in one way or another. It is a great honor and pleasure for me to do research under his supervision and command. I would like to thank all the faculty members of the Department of computer science and software engineering, who helped me in various ways, in the completion of my research work and thesis.

I am gratified to my parents who supported me a lot throughout my studies. With their devotion, steadfastness, and keen interest, it becomes possible that I have completed my degree of MS (CS) from International Islamic University.

I want to dedicate this thesis to my parents.

Muhammad Husnain Sardar
Reg#: 606-FBAS/MSCS/F10
Abstract

Classification is the data mining and pattern recognition technique used to classify the input patterns, instances on the basis of class label. Different classification models and methods are being used for this purpose. Some methods performed very well and some may give the average results. The use of ant colony optimization method for extraction of rules performed very well and produces incredible results than other techniques like decision trees, regression, and genetic algorithm etc. The use of fuzzy logic with the triangular based fuzzy linguistic function did not produce enough good results, because of the sharp or crisp edges. The decisions made by the system will be the crisp and tight. The replacement of the triangular fuzzy linguistic function with the trapezoidal fuzzy linguistic function by fuzzifying the input values in contact with the ant colony optimization for extraction of rules make possible to achieve the classification rate and classification accuracy better than the previous state of art method. The results generated by the proposed technique are comparable with the existing state of art methods that have been employed on the same particular diabetes disease dataset for the sake of classification. The results of proposed technique are the best in all respects as compare to result of existing state of art methods.
Table of Contents

Contents
1 Introduction .................................................................................................................................................. 9
  1.1 Motivation .............................................................................................................................................. 9
  1.2 Classification ....................................................................................................................................... 10
  1.3 Fuzzification .......................................................................................................................................... 16
    1.3.1 Fuzzifier ........................................................................................................................................ 17
    1.3.2 Inference Engine .......................................................................................................................... 17
    1.3.3 Defuzzifier ...................................................................................................................................... 18
  1.4 Ant Colony Optimization (ACO) .......................................................................................................... 18
  1.5 Contribution ......................................................................................................................................... 19
  1.6 Thesis Layout ....................................................................................................................................... 19
2 Literature Review .................................................................................................................................... 20
  2.1 ACO (ant colony optimization) based Techniques .............................................................................. 20
  2.2 Fuzzy logic based Techniques ............................................................................................................ 31
  2.3 Genetic Algorithm based Technique ................................................................................................. 38
  2.4 Summary ............................................................................................................................................... 40
3 Problem Statement .................................................................................................................................. 42
  3.1 Problem statement ............................................................................................................................... 42
  3.2 Research Objectives ........................................................................................................................... 43
4 Proposed Solution ..................................................................................................................................... 45
  4.1 Proposed Architecture ......................................................................................................................... 45
  4.2 Gini Fuzzy membership function ........................................................................................................ 47
  4.3 ACO based rule generation ................................................................................................................ 49
  4.4 Gini Index ............................................................................................................................................ 51
  4.5 Experimental results and Discussions ............................................................................................... 52
    4.5.1 Date set ......................................................................................................................................... 52
4.5.2 Performance Measures................................................................. 53
  4.5.2.1 Classification Rate................................................................. 53
  4.5.2.2 Precision and Recall ............................................................. 54
  4.5.2.3 F-Measure............................................................................. 54
4.5.3 Simulation Tool ........................................................................... 54
4.5.4 Results ....................................................................................... 55
  4.5.4.1 Numerical Results................................................................. 55
  4.5.4.2 Graphical Results ................................................................. 58
4.6 Summary ....................................................................................... 62

5 Conclusions and Future Work .......................................................... 63
  5.1 Conclusion................................................................................... 63
  5.2 Limitation.................................................................................... 63
  5.3 Future Work ................................................................................ 63

6 References........................................................................................ 64
List of Figures

<table>
<thead>
<tr>
<th>Figure #</th>
<th>Figure Description</th>
<th>Page #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>General Data mining process model</td>
<td>13</td>
</tr>
<tr>
<td>1.2</td>
<td>Fuzzy Logic system</td>
<td>17</td>
</tr>
<tr>
<td>2.1</td>
<td>Algorithm of state transition rules</td>
<td>26</td>
</tr>
<tr>
<td>2.2</td>
<td>Trapezoidal shape fuzzy membership function</td>
<td>32</td>
</tr>
<tr>
<td>2.3</td>
<td>Fuzzy classification boundaries</td>
<td>36</td>
</tr>
<tr>
<td>2.4</td>
<td>Classification problem</td>
<td>39</td>
</tr>
<tr>
<td>3.1</td>
<td>Triangular partitioned fuzzy membership function</td>
<td>42</td>
</tr>
<tr>
<td>3.2</td>
<td>Triangular fuzzy function</td>
<td>43</td>
</tr>
<tr>
<td>3.3</td>
<td>Trapezoidal fuzzy function</td>
<td>43</td>
</tr>
<tr>
<td>4.1</td>
<td>Architecture of proposed solution</td>
<td>46</td>
</tr>
<tr>
<td>4.2</td>
<td>Flow chart of gini fuzzy membership function</td>
<td>48</td>
</tr>
<tr>
<td>4.3</td>
<td>Rule extraction mechanism of ACO</td>
<td>50</td>
</tr>
<tr>
<td>4.4</td>
<td>Effect of number of ants on classification rate</td>
<td>59</td>
</tr>
<tr>
<td>4.5</td>
<td>Effect of number of ants to build the classifier</td>
<td>60</td>
</tr>
<tr>
<td>4.6</td>
<td>Effect of Max_Change parameter on classification rate</td>
<td>61</td>
</tr>
</tbody>
</table>
# List of Tables

<table>
<thead>
<tr>
<th>Table #</th>
<th>Table Description</th>
<th>Page #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3</td>
<td>Main categories of classification methods</td>
<td>11</td>
</tr>
<tr>
<td>4.1</td>
<td>Brief description of dataset</td>
<td>52</td>
</tr>
<tr>
<td>4.2</td>
<td>Subset of entire dataset</td>
<td>52</td>
</tr>
<tr>
<td>4.3</td>
<td>Classification accuracy obtained by different methods for PID</td>
<td>56</td>
</tr>
<tr>
<td>4.4</td>
<td>Classification rate obtained by different methods for PID</td>
<td>57</td>
</tr>
</tbody>
</table>
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Abbreviated Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACO</td>
<td>Ant Colony Optimization</td>
</tr>
<tr>
<td>DAG</td>
<td>Directed Acyclic Graph</td>
</tr>
<tr>
<td>TSP</td>
<td>Travelling Salesman Problem</td>
</tr>
<tr>
<td>ATSP</td>
<td>Asymmetric Travelling Salesman Problem</td>
</tr>
<tr>
<td>QAP</td>
<td>Quadratic Assignment Problem</td>
</tr>
<tr>
<td>JSP</td>
<td>Job Shop Scheduling Problem</td>
</tr>
<tr>
<td>FDT</td>
<td>Fuzzy Decision Trees</td>
</tr>
<tr>
<td>G-FDT</td>
<td>Gini-Fuzzy Decision Trees</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>FCS-Antminer</td>
<td>Fuzzy Classification System-Antminer</td>
</tr>
<tr>
<td>PID</td>
<td>Pima Indian Diabetes</td>
</tr>
<tr>
<td>TP</td>
<td>True Positives</td>
</tr>
<tr>
<td>TN</td>
<td>True Negatives</td>
</tr>
<tr>
<td>FP</td>
<td>False Positives</td>
</tr>
<tr>
<td>FN</td>
<td>False Negatives</td>
</tr>
<tr>
<td>MLNN</td>
<td>Multi-Layer Neural Network</td>
</tr>
<tr>
<td>QDA</td>
<td>Quantitative Descriptive Analysis</td>
</tr>
<tr>
<td>ASR</td>
<td>Automatic Speech Recognition</td>
</tr>
<tr>
<td>kNN</td>
<td>k-Nearest Neighbor</td>
</tr>
<tr>
<td>RBF</td>
<td>Radial Basis Function</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>GRNN</td>
<td>Generalized Regression Neural Network</td>
</tr>
<tr>
<td>LS-SVM</td>
<td>Least Square Support Vector Machine</td>
</tr>
<tr>
<td>NNGE</td>
<td>Nearest Neighbor Generalization</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Motivation

The motivation behind this research is to facilitate the doctor’s community and provide them an enough accurate system whose accuracy will be the best of all the previous versions that are so far designed for the classification of diabetes disease. The medical field has some discrepancies from the point of disease classification, the reason of it is much similarity between test results and symptoms shown in the reports of that particular test results. These discrepancies related to the early diagnose of the correct disease, because some diseases are complicated and dangerous, due to their late recognition a patient may suffer severity. Only a physician with good knowledge and with good experience can classify from the report of test results that from which disease the patient is actually suffering. This is all due to the fact that some disease are so much complicated that their symptoms are mixing up with the other disease, so at that stage physician have to diagnose the actual disease on his previous knowledge and practice. The wrong diagnostic may lead a patient to death because the actual disease which is still hidden and will start its worse effects on the body by spreading the gems in the whole body with the passage of time. So our work is on exploration of such type of disease known as “Diabetes Mellitus” and this particular disease is a type of diabetes disease. There are two types of diabetes disease: one is known as type-1 and second is known as type-2 or diabetes mellitus (our work is concerned to Type-2).

The aim is to propose a classification system for diagnosis of disease (Diabetes Mellitus) by using the combination of different techniques which can make a classifier more and more accurate with respect to other classifiers or systems that has been made in the past for the same purpose [5] [6] [7] [8]. The things on which more and more emphasized in the purposed technique are the classification and accuracy rates of classifier and their comparison of the classification results generated by the proposed technique with the results of other methods that have been developed.
specifically for this purpose up till now. The aim of particularly focusing the accuracy and classification rate is to point out the dangers, effects, hidden in the disease, which are unknown by an ordinary human. Actually the diabetes is a metabolic disease and one’s suffer form it when the glucose level in the blood exceeds and in the result production of insulin in the body minimized, whereas on the other hand the insulin is very important for body regarding to energy purposes. The diabetes disease helped to invoke other diseases like blindness, heart, kidneys, blood pressure and damage of nerves [7]. So we can name this disease as the silent killer. Diabetes itself has two types: type 1, and type 2 as discussed earlier. The most dangerous type is type 2 and it is also known by the name Diabetes mellitus [9]. Millions of human are suffering from this disease but they don’t aware of it till decades despite of the fact that our medical industry is advancing day by day to latest and more accurate methods of treatments and diagnosis. The difficulty to diagnose the type 2 (Diabetes Mellitus) disease is the symptoms shown by the patients. The patient have mixed symptoms which are part of other diseases also, so the doctor at that stage unable to identify exactly that the patient is suffering from diabetes type 2 disease. So physician needs to have full knowledge and expertise in this field so that he can diagnose this disease accurately. To reduce the physician effort and to minimize the chances of doubt, a classifier is being made by us which takes the patterns (tuple, record, instance) as an input (each pattern have set of attributes important to recognize the diabetes disease) and predicts [10] that whether the desire person is victim of diabetes disease or not.

1.2 Classification

Classification is an approach used to determine the valid class of pattern on the basis of characteristics of the given pattern [1]. In this technique we have data sets which have number of attributes depending upon the type of data and among these attributes there should be a one attribute named as Class Label. This class label attribute usually classify the data. Usually we build a classifier model by using any given technique of the classification and then trained our model on the given data set. When the process to learn an algorithm finishes then the model becomes ready
enough to predict or to classify the new samples by using the knowledge base that has been developed in the learning phase on the sample patterns. It can be used in any domain like Data Mining, Pattern Mining. There is lot of classification methods which can be used for classification purpose, all these methods are represented in Table 1.1.

<table>
<thead>
<tr>
<th>Group of methods with representatives</th>
<th>Model representation</th>
<th>Model evaluation</th>
<th>Model search</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inductive methods: Decision Trees: C4.5, CART</td>
<td>Decision tree (interpretable)</td>
<td>Probabilistic methods, penalty on complexity</td>
<td>Greedy</td>
</tr>
<tr>
<td>Coverage Rules: Decision Rules: PART, RIPPER</td>
<td>Decision rules: ranked disjunction of conjunctive rules (interpretable)</td>
<td>Classification error</td>
<td>Greedy</td>
</tr>
<tr>
<td>Bayesian methods: Bayes Net</td>
<td>Probabilistic, graphical (interpretable)</td>
<td>Likelihood, or posterior probability</td>
<td>EM</td>
</tr>
<tr>
<td>Linear methods: logistic regression, Fisher’s Linear Discriminant, Support Vector Machines</td>
<td>Separation hyper plane (non-interpretable)</td>
<td>Likelihood, error, geometric margin</td>
<td>SMO, method punctilio interior</td>
</tr>
<tr>
<td>Non Linear methods: Artificial Neural Network</td>
<td>Network of interconnected neurons, links with weights (non-interpretable)</td>
<td>Mean squared error, cross entropy</td>
<td>Gradient descent, conjugate gradient, EM, evolutionary, etc.</td>
</tr>
<tr>
<td>Lazy methods: K-Nearest neighbor</td>
<td>Implicit (in data): no explicit model (non-interpretable)</td>
<td>Classification error</td>
<td>External, non-specific to algorithm</td>
</tr>
<tr>
<td>Meta Techniques: Ensemble, bagging, boosting, wagging, stacking, multiple models techniques</td>
<td>Group of models, one for each base classifier (non-interpretable)</td>
<td>Specific to the base classifiers</td>
<td>Specific to the base classifiers, boosting</td>
</tr>
</tbody>
</table>

Table 1.1: The main categories of Classification Methods
Classification itself is the technique of data mining, in which we classify the patterns on the basis of class label. Data mining is the process to discover the knowledge. In it different methods are applied with the combination of artificial intelligence, machine learning to extract out the useful knowledge from the data set. The other name of this useful knowledge is information and it is in some structural form which makes understanding easy for everyone. In data mining before extraction of useful information there are series of steps or procedures to perform, so all those steps or procedures are shown graphically in the Figure 1.1 which represent the basic data mining model.
The brief explanation of each module of the Figure 1.1 is as.

**Input:** The input means that the entire data set which has to be processed to solve the entire problem. The dataset can be in the form of text, number or images depending upon the nature of problem. According to nature of our problem on which we have
decided to work is in numeric form. Basically this is the prediction problem in which we have to train our method according to the input values of dataset and then on the basis of these values our method will identify the new cases after predicting on the experience of prior knowledge that has been built in the database of the method in the learning phase.

**Feature Selection:** This term refers to the selection of characteristic or attributes of the dataset that are important to find out the solution. It also depends upon the type of dataset and nature of problem. Suppose there is dataset which have 300 attributes and out of these 300 attributes there are only 30 attributes which are useful to build the classifier for classification, then we would not consider the other 270 attributes and consider only these 30 attributes. It would not have any negative impact on the build classifier and their accuracy. There is possibility that other columns have noisy or redundant data. So to keep away from the meaningless data we use feature selection process.

**Technique Selection:** This term refers to the selection of technique according to type of problem. If problem is related to classification then there is lot of methods like C.4.5, ID3, Clustering, ACO etc. Every researcher tries to choose the technique which is useful, efficient and have the ability to give good accuracy.

**Data Pre-processing and Transformation:** The data pre-processing is actually the single name of four different things like data cleaning, data integration, data transformation, and data reduction. Simply in data pre-processing phase the aim is to remove the noisy, redundant, and inconsistent data from the data set. This is the particular step or procedure where we define the policies or rules for the records which have null values in the dataset.

**Training:** In this phase we train our method on the given dataset which is so called learning of the method. The process of learning keeps going on until it learn all the rules and start predicting the new cases on the base of learned knowledge.

**Testing:** The testing phase is the phase where overall performance of the build classifier or method is to test. If the testing phase produce good results than it is acceptable for decision making otherwise it would be not.
**Decision Making:** This is the phase where the implemented or developed method is ready to use in real sense or in actual application. Suppose if the dataset is related to diagnose of disease and if it successfully build and started prediction in a good manner then we can use it for the patient to diagnose that disease by entering the results of different particular tests which are necessary for diagnose of that type of disease.

The model presented in Fig.1.1 is an iterative model and very basic model of data mining approach. The process starts from the input. In other words we can say that we have dataset and this dataset as a whole is known as an input of the data mining model. After input there is further step known as feature selection, in feature selection we extract the features that are important and can covers the whole aspects of the model in all manners. The feature selection is valid for the datasets which have high dimensions. Some of the most common and core steps of data mining model are like (Technique-selection, data pre-processing, training, and testing). The technique selection means that which technique or method will be used to find out the solution for the given problem. Suppose the problem is related to classification then the techniques that can be used for classification are neural network, clustering, ant colony optimization, genetic algorithm, decision trees, regression etc. After technique selection there comes a phase of data pre-processing, in this phase the data should be made noise free and consistent. Moreover if there is redundancy then it will be removed. For all this a mathematical function or expression can be defined or it can be solved through the programming technique. When the data becomes noise free then it is ready to for training or learning phase. There are two types of learning.

1. Supervised learning.
2. Unsupervised learning.

In supervised learning the algorithm is given a labeled training dataset and on this dataset the algorithm learned and builds up its knowledge base. Whereas in unsupervised learning the algorithm is not provided a labeled training dataset, there is no labeled class on the base of which a method can decide or classify the pattern. After completion of leaning or training phase there comes a testing phase, in testing phase the algorithm is tested on the patterns which are unknown for it, the algorithm...
has to classify the patterns on the base of knowledge learned by it during the training. If the testing phase generates the satisfactory results then the build method or classifier is ready to use for the decision making in real life. The testing phase is the most important and critical phase, all the efforts did in the previous phases are relying upon this. If the testing phase failed to produce the satisfactory results then it means that the choice of technique is wrong so we have to choose another technique to solve the problem. It will continue and repeats the process until the testing phase generates the satisfactory results.

1.3 Fuzzification

Fuzzification is a method through which crisp values transformed in to the member of grades and represented in linguistic terms which is called fuzzy sets. For association of grade in terms of relevant linguistic term we use membership functions. So in this way a real scalar value transformed in to the fuzzy value. Usually three type of fuzzifiers used for Fuzzification. These three fuzzifiers are singleton fuzzifiers, Gaussian fuzzifiers, and triangular or trapezoidal fuzzifiers. Fuzzy logic has great competency to handle the problems which have uncertainty and vagueness [2] [3] [4]. The fuzzy logic has become a well-known solution for problems like classification it is being variously used. The Zadeh [5] use the fuzzy logic first time for the problems of classification. Fuzzy logic basically works on the if-then rules and these rules constructed by seeing the nature of data which is to be fuzzified. These if-then rules are very simple and every human can understand by seeing the rules that what the classifier actually wants to do and provide deeper vision into the classifier and as well as decision making process [2]. The interpretation of results is very easy and simple. The Figure 1.2 shows that how fuzzy logic system works and implemented.
The fuzzy logic system has following three main components.

1. Fuzzifier
2. Inference
3. Defuzzifier

### 1.3.1 Fuzzifier

The role of fuzzifier is to convert the crisp values or set of crisp values in to fuzzy values or set of fuzzy values and then after conversion these values will be passed from the inference engine. For conversion of crisp values into fuzzy values, the fuzzy functions will be used and these functions are not built-in. The researcher or developer itself defines the function according to the given data. The result of fuzzy values is to increase the accuracy power of the method.

### 1.3.2 Inference Engine

After conversion of crisp values in to fuzzy values, these values will pass through the inference. The basic working of the inference engine is modification of the fuzzy membership function and to gain more access on the input handling to produce good output. The inference engine has the rule base where rules are defined for the
modification of fuzzy values. The rules are in the form of If-Then which are very easy and simple to understand.

1.3.3 Defuzzifier

Defuzzifier is the inverse of fuzzifier in a way that it converts the generated fuzzy values by inference in to crisp values again. Actually we use defuzzifier to check that how much fuzzification effected the values in actual.

1.4 Ant Colony Optimization (ACO)

Ant colony algorithm introduced by the Marco Dorgio in 1992. ACO-based algorithms are stochastic by nature and have the ability to copy the actual conduct of ants and their colonies. ACO algorithm used to find the shortest path from source to destination or food. The ACO is also used to generate the rules in real time problems. Discovery of the shortest path by the ants from source to food is actually the rules which are named as paths or trails. If there are lots of solutions exist in the problem then ant colony algorithm helps to find out the solution in terms of the path which will be a shortest one. ACO imitates the behavior of real ants in a way that the ants while travelling towards the food or destination pledge a chemical substance called pheromone [12]. This pheromone helps other ants to smell and to travel on the same path [11] because the ants have no eyes and they can travel only by smelling the pheromone. The pheromone keeps on evaporating with the passage of time. Initially all the ants will not follow the shortest path, some ants may follow the longer path and as a result they will take more time to reach to food source, so on the other hand the pheromone level becomes low and low due to evaporation on such particular path. In the result the path which is longer and have low pheromone level will become less attractive for other ants. So a time will come when the ants with paths other than shortest will left those paths and will likely to follow the shortest path because of high level of pheromone [11], in a result the whole colony of ant will follow that path [12]. At initial level pheromone level will be zero so the ant selects the path arbitrarily. The shorter the path, the more will be the pheromone on such path and more ants will attracted by that path. The pheromone evaporates with the
time. So the other paths which are longer will not be followed by the ants because the pheromone levels will be low as compared to the shortest.

1.5 Contribution

The novel method used for the classification of diabetes disease is the combination of Gini Index fuzzification technique with ant colony optimization technique. The gini index applied here is to fuzzify the values, whereas the ant colony optimization technique will make use of fuzzified values for the extraction of rules. Previously ant colony optimization is applied with contact of fuzzy sets. The partition of fuzzy sets are uniform without use of any quantitative approach, moreover the fuzzy linguistic term used as representation of fuzzy sets is triangular. The triangular fuzzy membership function comprises of sharp edges or crisp boundaries which may leads to misclassification of patterns because the decisions made in triangular fuzzy representation are crisp which affect the classification rate of the classifier and its performance. In other words these are not realistic in nature because only categorical attributes can be used here. So we have tried to omit these categorical attributes by continuous attribute by applying the gini index on each attribute’s value of dataset, in the same way the triangular fuzzy membership function replaced by the trapezoidal fuzzy membership function which claims that the classification will be improve and this claim will come to true later in experiment phase.

1.6 Thesis Layout

The detailed related work in the form of literature review is represented in chapter 2. In this chapter all the versions of ant miner and different fuzzy if –then rules base classification systems have been discussed. The chapter 3 comprises of problem statement in detail. The chapter 4 is about the derived proposed solution to solve the stated problem statement and their results in the form of figures and tables as well. The Chapter 5 explains the conclusion of the entire proposed approach, limitations involve in the proposed approach and future work. In the end we have the references.
2 Literature Review

A huge amount of research is focusing the two major areas of computer science like pattern recognition and data mining. Different methods and techniques have been evolved for the classification, rule learning and as well as the rule extraction processes. The involvement of importance of fuzzy logic in rule association or classification problems cannot be rejected at any cost. The use of fuzzy logic has opened a new era through which the results and system’s performance can make more and better. This chapter comprises of the related work to our novel approach.

The literature review has divided into three sections like the section 2.1 is related to the approaches that have used the ant colony optimization method for classification problem by extracting the rules. This section will not cover all the state of art methods, the few methods that are closest to novel approach are explained here. The section 2.2 referred as fuzzy based techniques. In it quite a few research papers have discussed in which fuzzy logic has been applied in contact of other techniques. The section 2.3 express the technique genetic algorithm used that have been used for the classification through extraction of rules.

2.1 ACO (ant colony optimization) based Techniques

The ant colony optimization technique is very well known to solve the classification problems by extracting the rules from the dataset. The reading and research material related to ant colony optimization and their usage is huge in volume but we will discuss quite a few which are helpful and become the basis of our novel approach. The mentioned research papers are related to classification problems. It is also mentioned that how and who use the ant colony optimization first time for extraction of rules, by whom the ant miner discovered and how the various versions of ant miner evolved.

The very first novel approach that has been used for classification through extraction of rules by using the ant colony optimization is referred as data mining with the use
of ant colony optimization algorithm [13]. Data mining is the extraction of knowledge, and this knowledge should be accurate and graspable for the user. The reason of its grasp ability is that, the extracted knowledge has to be uses by the human for decision making processes and this decision making process is the fruit of data mining for which the methods and techniques are built. In data mining there are number of methods which are used for clustering, classification, regression etc. The use of each method is dependable upon the type and nature of problem. Ant colony optimization used for clustering in the data mining. No one knows that it can also be used for the classification and this area was unexplored before this research. In this current explaining approach [13] the ant colony optimization first time used for the classification by extracting the rules and named it as Ant-Miner. Insects like Termites, Bees, Ants is capable to perform its own task independently, but when they cooperate with each other than the whole colony can do the difficult task like food searching and its storage without any supervision and central command and this ability referred to as Swarm Intelligence. The inspiration to discover the ant miner is that the ants are intelligent and have enough ability to find the shortest path from source to food. Genetic Algorithm also used for rule discovery but the rules discovered by the genetic algorithm are not qualitative with respect to rules discovered by the ant miner. The reason behind this is ant miner used pheromone updating strategy which says that always choose that path on which pheromone level is abundant and that path will surely the shortest path among the all. The ants in it are artificial so the Equation 2.1 explains how the pheromone table is initialized with the pheromone.

\[
T_{ij} (t = 0) = \frac{1}{\sum_{i=1}^{a} b_i} 
\]

Where ‘a’ is used as total number of attributes in the data set and \( b_i \) shows the values that can be taken from the given attribute.
Each ant discovers the rule but the question is which rule is the best to classify the
given pattern. So in ant miner a function is defined which is called heuristic function.
This function is capable to precisely define that the how much pheromone level is
available on the constructed rule. So the rule with highest heuristic value is the sign
that the quantity of pheromone level is high. The heuristic function is defined as in
Equation 2.2.

$$\eta_{ij} = \frac{\log_2 k - H(W \mid A_i = V_{ij})}{\sum_{i=1}^{a} x_i \sum_{j=1}^{b} (\log_2 k - H(W \mid A_i = V_{ij}))}$$

(2.2)

Where

$$H(W \mid A_i = V_{ij}) = -\sum_{n=1}^{k} P(\omega \mid A_i = V_{ij}). \log_2 P(\omega \mid A_i = V_{ij})$$

‘a’ shows the total number of attributes available in the dataset.

$x_i$ depends upon $A_i$, if $A_i$ used by the ant then $x_i$ value will be set to 1, otherwise
its value will be zero.

$b_i$ is the number values for the $i$th attribute.

Moreover ant miner is quite different from the Decision Trees and Evolutionary
Algorithms. In [13] this technique compared with the CN2 [14] [15]. CN2 is used for
classification rule discovery process. The rule discovery process is different from the
ant miner. CN2 uses the beam search process for rule discovery. The working of
beam search is that all the all the possible terms added in the current partial rule,
evaluate this partial rule and return the best partial rule. Beam width refers to as
where all the best partial rules retained and it returns the best of all others will be
omitted. So CN2 is different from ant miner in the sense that there is no mechanism
to achieve the quality of rule like pheromone updating in the ant miner. The other
difference between the CN2 and the Ant miner is that CN2 is deterministic approach
whereas the ant miner is the stochastic approach. Both CN2 and the Ant miner are
tried upon the six public data sets (Ljubljana breast cancer, Wisconsin breast cancer,
dermatology, tic-tac-toe, Cleveland, hepatitis, heart disease) with respect to performance and these data sets are taken from the Irvine Repository. The performance of the Ant miner with respect to prediction accuracy is much better than the CN2 in four data sets, whereas in one data set predictive accuracy is same and in the one data set CN2 is better than the Ant miner. Ant miner discovered simple and much smaller rules than the CN2. In Data Mining the consideration of rules simplicity is very important because of predictive accuracy and as well as it is easy to understand for the human. There are some limitation which have been emerged in the Ant miner that particular emerging limitation is the use of Entropy function which increased the overhead of the calculations. It is used only for categorical attributes and not suitable for the discrete attributes.

Ant miner discovered and applied successfully on the dataset to solve the classification problem, but the computation time taken by the ant miner is too much, because it based upon the entropy function in which logarithm is applied. So the ant miner is revised in a way that the simplest function (density based heuristic function) will be defined instead of entropy, as a result the computation time decreased significantly and named it as ant miner2 [21]. The ant miner2 is the name given to it just because it is based upon the ant colony optimization technique which imitates the behavior of real ants. Parpinelli first time used the ACO algorithm for classification and named it Ant-miner [13]. The ant-miner working inspiration is of ACO system. The ants are blind and they know nothing but the thing through which these ants travel is the chemical substance excreted by the ants (pheromone) and deposit on the path while they walking. This pheromone helps other ants to find a path from source to food. The path which has the more pheromone that will be the shorter path and whole ants later will follow that shorter path. With the passage of time the shorter path have pheromone in abundant quantity due to more deposition and less evaporation of pheromone and this effect is called autocatalysis. Ant-miner [13] is enhanced with some changes in the heuristic information and names it as Ant-miner2 [21]. The heuristic information calculated in the ant miner is based upon the entropy function in which too many calculations in terms of multiplication and division involved. Moreover logarithm is also calculated to calculate the entropy
value. So to overcome this computational and calculation overhead a density based heuristic function is used which is inherited from the Bayesian classification. In this function there is no entropy and no logarithm, it is a simpler function with one division only. This simpler heuristic function is defined in Equation 2.3.

\[ \eta_{ij} = \frac{\log_2(k) - \text{Info}_{ij}}{\sum_i \sum_j \log_2(k) - \text{Info}_{ij}} \]  

(2.3)

Where

\[ \text{Info}_{ij} = -\sum_{w=1}^{k} \left[ \frac{\text{freq}_{T_{ij}^w}}{|T_{ij}|} \right] \ast \log_2 \left[ \frac{\text{freq}_{T_{ij}^w}}{|T_{ij}|} \right] \]

\( k \) denotes the total number of classes in the dataset.
\(|T_{ij}|\) represents the number of cases in \( T_{ij} \) partition.

\( \text{freq}_{T_{ij}^w} \) represents the number of cases in \( T_{ij} \) partition with class \( w \).

‘\( a \)’ is the total number of attributes in the dataset.
\( b_i \) is the number values in attribute \( i \) domain.

As the higher the value of \( \text{Info}_{ij} \), there are very less or fewer chances that the ant will choose the term to construct the rule. The reason of it is that \( \text{Info}_{ij} \) is inversely proportion to the heuristic value.

Actually the concept behind is accurate heuristic value is not needed here because the pheromone is already initialized at the first step so it would not cause the serious effect on the accuracy. The accuracy rate of ant miner2 is the same as of ant miner but the computation overhead is minimized. The ant miner2 takes less time in processing as compared to ant miner. The new proposed function in ant miner2 does not affect the overall performance of the rules and the system because there are further steps like rule pruning, pheromone updating and these steps are the same as of ant miner. The proposed technique applied on the data set Wisconsin Breast...
cancer taken from the UCI machine learning repository the results of ant miner and ant miner2 are the same the only difference is of time required in execution or the time in which processing completed. In ant miner2 [21] limitations to handle only categorical attributes still existed like ant miner [13].

In ant miner [13] and ant miner2 [21] ant colony optimization method is applied for rules extraction to solve the classification problem, both methods perform well to some extent. In ant miner and ant miner2 the problem extract out that there accuracy and classification rate can be increased if a pheromone updated strategy is applied. This approach implemented later on and named it as ant miner3 [22]. In the previous versions of ant miner which we have discussed above, the selection of terms by ants to construct a rule is done through the pheromone level which is initialized at the start to all the cells by a function and the heuristic function’s value. The heuristic value is used to know about the prediction power of the term added in construction of rule. The ant constructs a rule by adding the terms. After rule construction the pheromone level for that rule increased whereas the value of heuristic function remains unchanged. When the new ants came for rule construction then they incline to choose the terms which have used in the previous rule because of high pheromone level, the unused terms remains unused. As a result whole ants will converge to a single path very quickly. It leads to failure because alternative paths are didn’t use by ants. It may possible that from alternative paths there will be a shorter path. In this paper a new pheromone updating strategy introduces and called this version to be known as antminer3. The functionality of rule construction will be the same as explained in earlier versions of ant miners but after rule construction the pheromone updating strategy for the terms that are used in the rule construction are determined by the newly purposed function. This newly proposed function is expressed in the Equation 2.4.

\[
\Gamma_{ij}(t) = (1 - \rho) \cdot \Gamma_{ij}(t-1) + \left(1 - \frac{1}{1+Q}\right) \cdot \Gamma_{ij}(t-1)
\]  

(2.4)
Where

\( \rho \) is the rate of pheromone evaporation.

\( Q \) represents the quality of discovered rule.

The pheromone of the terms which are not used in the rule construction also updated through normalization process. The tradeoff between the pheromone evaporation rate and the quality of constructed rule is also maintained. Like the terms which have high pheromone level but in rules these terms not used then their pheromone level will have to decrease due to evaporation. As we know that the ants are intelligent and cooperative with each other in their colonies. The ants have no memory and they could only cooperate with the help of environment. The concept of exploitation is expressed as the pheromone located on the edges in ant colony system for the sake of distributed memory for the ants. As the ants have no memory so they communicate with each other through this pheromone and this will become the way of indirect communication. This is called the exploitation because the ants will likely to move on the path which has more pheromone and it will increase the probability to choose the terms which are used in the previous rules. The concept of exploration vanished. To make the use of exploration concept the state transition rules are applied. The transition rules working in the form of algorithm is defined in the Figure 2.1.

\[
\begin{align*}
\text{if } q_1 \leq \varphi \\
\text{loop} \\
\text{if } q_2 \leq \sum_{j \in J_i} P_{ij} \\
\text{Then choose term}_{ij} \\
\text{End loop} \\
\text{Else} \\
\text{choose term}_{ij} \text{ with max } P_{ij}
\end{align*}
\]

Figure 2.1: Algorithm of state transition rules
Where

$q_1$ and $q_2$ are any random numbers.

$\varphi$ is the parameter in the internal $[0,1]$.

$j_i$ is the value of $i$-th attribute.

$P(j_i)$ is the probability that tells by calculating whether this term will be added in the current rule to being constructed or not.

The propose technique implemented on the various public data sets like tic-tac-toe and Wisconsin breast cancer, taken from the UCI machine learning repository and results are then compared with the ant miner. It has been noticed that the accuracy rate of ant miner3 is increased a lot. Moreover the numbers of rules are more constructed in the ant miner3 as compared to the ant miner.

The versions of ant miner keep on advancing with the passage of time and could not stop on the ant miner3. The reason of it is the hunger of getting the more and more accuracy in classification. As discussed earlier the ant miner [13], ant miner2 [21], ant miner3 [22], evolved with changes in existing mathematical functions or by addition of more functions that would be beneficial in terms of classification accuracy. The new version ant miner+ [23] emerged with the changes applied in the environment where the ants will travel. This changed environment will be the graphical or in the form of graph. As discussed earlier the versions of ant miners like ant miner [13], ant miner2 [21], ant miner3 [22] with a differences of techniques used and approaches to solve a problem, same in the ant miner3 [23] a new techniques a new way to solve problems of classifications have been introduced.

The proposed environment in ant miner3 is the graphical and named it direct acyclic graph (DAG). For better construction of the graph the pheromone and heuristic functions defined with weight parameters. The max-min ant system has proposed, moreover the early stopping criteria have been introduced in the ant miner3. The working of ant miner3 is like that at first the direct acyclic graph constructed, the graph have the start and as well as the stop vertex. Each ant will take start from the star vertex and will move in the graph to construct the rule until it reached the stop
vertex. The ant with shortest path will update the pheromone. Meanwhile pheromone evaporation also increases on the edges of the graph. Pheromone level will be handled by the max-min ant system. According to proposed technique the method will converge to system when the edges of the path defined by the ant has the maximum pheromone level and all the other edges in the graph have pheromone level to minimum. The probability of any ant to choose the edges in the graph is totally dependent upon the pheromone quantity and the heuristic value of the edge in the directed acyclic graph (DAG). As each edge has the pheromone so the probability of ant movement from one edge to other edge is given in Equation 2.5.

\[
P_{ij}(t) = \frac{[\tau_{(v_{t-1,k,v_{t},j})}(t)]^\alpha \cdot [\eta_{v_{t},j}(t)]^\beta}{\sum_{l=1}^{p} [\tau_{(v_{t-1,k,v_{t},l})}(t)]^\alpha \cdot [\eta_{v_{t},l}(t)]^\beta}
\]

(2.5)

Where \(\alpha\) and \(\beta\) are the weight parameters used to obtain the importance of pheromone and heuristic function as well.

\(\eta_{v_{t},j}\) is the heuristic function or heuristic value which is expressed in Equation 2.6.

\[
\eta_{v_{t},j}(t) = \left| \frac{T_{ij} \& CLASS = \text{class}_{\text{ant}}}{|T_{ij}|} \right|
\]

(2.6)

Where \(\text{class}_{\text{ant}}\) represents the heuristic value is dependent upon the class chosen by the ant.

The heuristic function used in antminer3 is different from the previous versions in the sense that it gives the right search direction and this is the fact that the ants know during the phase of rule construction that to which class exactly they are extracting the rules. We can say that rules construction is specific to class in the ant miner3. In pheromone evaporation phase the two terms are discussed one is evaporation and the other is reinforcement. Evaporation of the pheromone leads to the paths which are lesser used by the ants and reinforcement of the pheromone leads to the paths that are shorter and most of the ants used them. In rule pruning all the irrelevant terms that
are not too much useful in the extracted path are omitted. One best thing that is implemented in the ant miner3 is the early stopping criteria. As each data set holds the noisy data in it so to deal with that type of data a separate training set has been made from that data and then it is tested during the training session. At the time of test the error rate also measured, if the error rate increasing then the learning stopped and it means that the data set is noisy and it will give not give good accuracy rate during classification. The disadvantages of the ant miner3 are as it takes more time in computation. Moreover like other versions of ant miner it is also used for categorical values, the continuous values are discretized at first. The ant miner3 applied on various public data sets and then it is noticed that the accuracy rate and the rules generated by the antminer3 are very good as compared to other versions of ant miners.

The knowledge extracted from the versions of ant miner is enough to understand that how ants travel to construct the rule and how its efficiency can be increased. To know that how the whole ants cooperate with each other in moving the whole colony to the destination or food source can be known by the novel research ant system optimization by colony of cooperating agents [24]. As ant system is proposed by the inspiration of the ants, that they live together and can do a work by cooperating with each other without any involvement of any leader or planner. Similarly in the ant system the proposed method is used for the problems like stochastic, greedy, and for optimization techniques. The silent and vital properties of the proposed system are positive feedback, distributed computation, and constructive greedy heuristic. Positive feedback guarantees that the best fit solution will be emerged. Distributed computation property refers that the untimely conversion will not possible. Greedy heuristic property work as to find adequate solution of the problem in early phase of search. The proposed system is flexible in nature and can be used for the problems which have similar nature like TSP (travelling salesman problem) and ATSP (asymmetric travelling salesman problem). It is robust, means that this method with fewer changes can be used for the solution of other optimization problems like QAP (quadratic assignment problem) and JSP (job-shop scheduling problem). Same like the ant as their population whole shift to a path which is shorter this method is also
having same population base criteria of working. The search methodology is autocatalytic in behavior. The autocatalytic behavior means that the ants will gradually follow the path which is rich in pheromone and this path or trail will be the best solution. The other paths will be omitted. To save from the quick convergence the interaction of ants with each other is very important so that they can give a solution which is best and this is the goal of the proposed method. The advantage of the proposed method is computational efficiency with respect to previous stated methods. The results of this method are promising.

The last novel approach of our literature review for the section of ant colony base optimization techniques is the induction of fuzzy classification system via evolutionary ACO based algorithm [35]. The proposed technique involves the discovery of evolutionary algorithm which initiates the fuzzy classification rules. To achieve the target of best classification results our algorithm will use the ant colony optimization technique as a local searcher and in this way the final quality of final fuzzy classification system improved. The proposed algorithm is tested upon the dataset related to intrusion detection and it is classification problem with high dimensions. In the computer intrusion are the main cause of quality problems. today’s there are number of intrusions in each system and their growth is very fast because there are many hacking software available, moreover there is chance it may come from the web pages while using the internet. With respect to security point of view, lot of intention is being paid to this problem and different methods are developed in machine learning and data mining to cope with such kind of problems. Intrusion detection system use two approaches like anomaly detection and signature recognition. The patterns of signatures are stored in the signature recognition technique. These patterns compare the signatures with the stored ones. It detects the intrusion if any signature pattern matches with the stored ones. Actually the algorithm is trained according to the stored pattern. The concept of machine learning is involved. This type of intrusion detection is used is commercial based intrudction detection systems. the anomaly detection technique detect the intrusion when the system activities are not normal and changed with respect to daily life, like if there is automatic change comes in the profile of the computer. The aim of the proposed
technique is to recognize the signature intrusions from the computer network, and the name of technique which we used is fuzzy genetic learning. Genetic algorithm plays a vital role in extraction of the fuzzy rules and the compactness of the fuzzy rules as well.

Genetic algorithms are very useful for the classification problems. Michigan and Pittsburgh introduced the iterative learning method. According to this method each chromosome individually constructs the rule and adds it to the discovered rules list. A rule will be constructed in each iteration. So to make the base of Michigan approach we extended this approach and applied it on the classification problem which has five classes. For this purpose the genetic algorithm is used with combination of ant colony optimization heuristic. This combining technique is known as hybrid technique and it increase the search power of main algorithm locally and globally.

2.2 **Fuzzy logic based Techniques**

Fuzzy logic heavily used in the areas of data mining, pattern recognition, and in image processing for restoration of images or removal of noise from the image and much more. Fuzzy logic used in contact with other techniques or methods, the advantage of using the fuzzy logic is we can fuzzify our values and represent the results in terms of fuzzy linguistic terms in which values are representable in the form of partitioned fuzzy sets. The partitioning of fuzzy sets based upon the nature of problem, it can be uniform or it can be dynamic. The major and difficult thing in fuzzy logic is to define the fuzzy membership function by considering the values of whole dataset. Some of the techniques that are helpful for implementation of our novel approach are discussed in this section.

The fuzzifying gini index base decision trees [16] is the technique which helps how values can be fuzzified and how these values can be expressed in the form of trapezoidal fuzzy linguistic function. Classification is technique that is used to obtain the useful knowledge form the huge volume of data. Any classification model evaluated on two criteria like efficiency and clarity and this model shows the qualities of data in a good way and it is easy to understand. ID3 [17] and C4.5 [18]
[19] are entropy based classification models whereas the SLIQ [20] is also the classification model but it used gini index for split measure. The decision trees constructed by ID3, C4.5, SLIQ algorithms are the crisp decision trees. The trees with the crisp decision boundaries are unable to give maximum level of accuracy because the decisions are very tight and it is not natural to human. The reason behind this is if an unseen pattern with the little changed feature value comes than the built classifier will not classify that pattern. FDT (fuzzy decision trees) used to remove the randomness, ambiguity, vagueness in the data. Fuzzy decision trees give good accuracy but the problem is to define the fuzzy function according to the data and nature of problem. There are no built in fuzzy functions so this process is done by the researcher itself. Fuzzy decisions trees are the mixture of fuzzy sets and decision trees. The technique used in the current paper is that to fuzzify the decision boundaries of the tree during their constructions whereas in previous approaches data pre-fuzzify first and then use that data in tree construction. Gini index is used as a split measures rather than the use of gain ratio, info gain, and fuzzy entropy. Fuzzy decision tree methods fuzzify the whole data set and convert all the values into some fuzzy language based on the fuzzy membership function and that membership function can be triangular or trapezoidal. Figure 2.1 shows the fuzzy linguistics that is taken after fuzzifying the data set.

![Trapezoidal fuzzy membership function](image)

Fig 2.2: Trapezoidal fuzzy membership function.

In fuzzifying gini index based decision trees the data taken attribute wise and then select the split point from the given data. The split point will be that where the class
variable changing like 0 to 1 or 1 to 0 if there are two outcomes of class variable. After finding the split point there will be values which are greater than the split point and smaller than the split point. Now on these values the defined fuzzy membership function will be applied. The fuzzy membership function used in the proposed technique is defined in Equation 2.7.

\[
\text{fuzzy value} = \begin{cases} 
1 - 1/(1 + \exp(-\sigma \cdot (val - \text{splitpoint}))) & \text{if } val \leq \text{splitpoint} \\
1/(1 + \exp(-\sigma \cdot (val - \text{splitpoint}))) & \text{if } val \geq \text{splitpoint} 
\end{cases}
\]  

(2.7)

The G-FDT (gini fuzzy decision tree) approach tested upon the various data sets taken from the UCI machine learning repository, this technique generated excellent results as compare to fuzzy decision tree method SLIQ. The time taken by the G-FDT is very less as compared to SLIQ. If we discuss the accuracy rate of the G-FDT then we will come to know that it is fairly good with respect to others.

The limitation of fuzzy logic to deal with the dataset which has minimum number of attributes has been overcome by the Dorgio and Stutzle in the performance evaluation of fuzzy classifier systems for multi-dimensional pattern classification problems [25]. The stated classifier system is simple because the fuzzy sets are expressed in terms of fuzzy linguistic terms base on the fuzzy membership high performance for the classification problems that have many attributes with continuous functions which are constant. Moreover the heuristic function used to find the resultant class and to calculate the grade of certainty of if-then rule is very simple. The proposed classifier is so simple that there is no need to change the fuzzy membership function, and as it is stated earlier that the each fuzzy if-then rule is expressed in terms of fuzzy language so it very easy to understand the rule and it provides deeper insight into the classifier for the human. So we can say that comprehensibility wise this system is very good. The proposed classifier generates better results with respect to other classifiers [26] [27]. Before this proposed technique it is assumed that the fuzzy rule base classifier with grid partition is feasible only for those problems which have less number of attributes, but this proposed method proved that those assumptions are false with the results on
applying different data sets that have continuous attributes and attributes of all the datasets are above then ten in numbers. In this state of art method the proposed fuzzy classification system, the fuzzy IF-Then rules that are used for the classification problem with c-class patterns are in n-dimensional space of pattern \([0,1]^n\). The general rule is described as in Equation 2.8.

**Rule** \(R_j: if \ x_1 \ is \ A_{j_1} \ and \ ...\ and \ x_n \ is \ A_{j_n} \)

then \(Class \ C_j \ with \ CF = CF_j \) \hspace{1cm} (2.8)

Where

\(R_j\) represents the fuzzy if-then rule on \(j_{th}\) location.

\(A_{j_1},...,A_{j_n}\) express the fuzzy sets with all predecessor terms.

\(C_j\) is the resultant class.

\(CF_j\) is the certainty grade for the \(j_{th}\) fuzzy if-then rule for \(R_j\)

The Equation 2.9 shows the function to find the certainty grade.

\[
CF_j = \frac{\beta_{class1}(R_j) - \beta_{class2}(R_j)}{\beta_{class1}(R_j) + \beta_{class2}(R_j)} \hspace{1cm} (2.9)
\]

Where

\[
\beta_{classh}(R_j) = \sum_{x_p \in classh} \mu_j(x_p), \hspace{1cm} h = 1, 2
\]

Let us suppose that there is chance that \(\beta_{class1}(R_j) = \beta_{class2}(R_j)\) then the consequent class \(C_j = \phi\) and then the certainty grade \(CF_j = 0\), because there is no class for which we can find the certainty grade.

The winner rule of this proposed technique can be finding out by computing the function defined in the Equation 2.10.

\[
\mu_j(x_p).CF_j = Max\{\mu_j(x_p).CF_j \mid R_j \in S\} \hspace{1cm} (2.10)
\]
The main theme related to high performance and clarity of rules is fulfilled by it. The limitation of this classifier is that it can’t minimize the fuzzy if-then rules. Moreover the quality of fuzzy rule is also not taking into account by the use of fitness function. To minimize the if-then rules is done by multi-objective genetic algorithm [28] using the Pittsburgh technique, minimization of if-then rules means that suppose ten rules are generated but if 5 rules out of the ten are more general and it predicts the whole data set then we will consider these five rules instead of the ten. Pittsburgh approach has one disadvantage which is related to computation time. It takes more time in computation with respect to our fuzzy classifier system.

The third one technique related to fuzzy logic is the adaptive fuzzy rule based classification systems [29]. An adaptive approach is presented here in this paper in the construction of fuzzy rule based classifier system to solve the classification problems. The working of proposed system is comprises of two procedures.

I. Error correction base learning.

II. Additional learning procedure.

The error correction base learning works in manner that it increase or decrease the grade of certainty of a fuzzy rule based upon the conditions. The certainty grade of any fuzzy rule has to decrease if the extracted rule misclassifies the patterns. Similarly the certainty grade increases when an extracted fuzzy rule correctly classifies the patterns. This is all done to prevent the classifier system from the rules which leads to misclassification. The working of additional learning process is omit the rules which are no longer needed, in other word can say that rule pruning will be take place. The limitations or the things that we have to take care in this presented system are to represent the fuzzy sets in any fuzzy lingual term, it is should be necessary to specify the fuzzy partition for pattern space at the start. There is no need to amend the fuzzy membership function for every antecedent fuzzy set. Using the adaptive fuzzy based classification technique, the following methodology used for determination of the classification boundary. Consider the under mentioned unidimensional fuzzy rules in Equation 2.11 and Equation 2.12 respectively.
$R_1 : \text{if } y \text{ is } B_1$

\[
\text{then } y \text{ belongs to Class } 1 \text{ with } CF = CF_1
\]  

(2.11)

$R_2 : \text{if } y \text{ is } B_2$

\[
\text{then } y \text{ belongs to Class } 2 \text{ with } CF = CF_2
\]  

(2.12)

Where $B_1$ and $B_2$ are fuzzy subsets.

The Figure 2.3 depicts the fuzzy classification boundary determined by the above rules.

![Fuzzy classification boundary](image)

Figure 2.3: Fuzzy classification boundaries

The line with dots in fig.2.2 is showing the membership function of $B_1$ and $B_2$. The solid lines are showing the product of membership functions and their related grade of certainty. The Equation 2.13 determined the classification boundary.

\[
\mu_1(y) \cdot CF_1 = \mu_2(y) \cdot CF_2
\]  

(2.13)

Where $\mu_1(y)$ and $\mu_2(y)$ are membership functions of $B_1$ and $B_2$ respectively. The ratio of $CF_1$ and $CF_2$ define the classification boundary.

The performance of pruning is not good enough that we can compare our number of extracted rules with the rules that are generated and pruned by the GA method.
Moreover the presented system will not be beneficial for the data sets that have lot of attributes with the numerical values, because the number of rules will be increased with the exponential power and it will be very difficult to handle. However the data set having number of attributes over ten with the numerical values can be handled by the approach [25] as describe previous.

For classification system we can find the efficient rules that can classify the data efficiently with the use of fuzzy logic. This methodology is discussed and mentioned published with the name “efficient fuzzy rules for classification” [32]. Fuzzy rules are very efficient to deal with the data which have uncertainties and vagueness. It is more like human because the fuzzy rules are very simple and very easy to understand for any human. Moreover the fuzzy rules are accurate, efficient and comprehensible. Here the fuzzy approach and genetic algorithm are both applied to handle the quantitative data. The proposed approach stated that the decision tree will be constructed against the membership functions, and then evaluation of rules carried out with respect to the accuracy and complexity. Classification methods like ID3, C4.5 are used to construct the decision tree for classification problems but there is one issue that they have no ability to handle the quantitative data. For quantitative data or quantitative attributes the crisp decisions are made which are not more like to natural and have less accuracy. To overcome this problem fuzzy approach is used so that quantitative data can be handled easily and it is more like natural. The only difficult thing in fuzzy approach is to define a fuzzy membership function according to the nature and volume of data set. This defined fuzzy membership function used to fuzzify the data and as well as to it represents the data into linguistic terms. In proposed approach the genetic algorithm is used to find out the best fuzzy rule set from set of fuzzy sets, moreover the decision tree will be constructed against the defined fuzzy membership function and it will be used to determine the accuracy of the classifier and as well as the complexity of the rule. The difference between the normal decision tree and fuzzy decision tree is both have nodes, arcs for representation of attributes and their values, but in fuzzy decision tree each arc have association with fuzzy linguistic term which is denoted by fuzzy set. The steps of proposed algorithm to construct the fuzzy decision tree through genetic approach are.
1. Produce the population for chromosomes using membership function at initial level.

2. Making the use of fuzzy membership function, construct the fuzzy decision tree.

3. Check the evaluation criteria for individual set of membership function correspond to chromosome by performance and complexity of constructed FDT.

4. Check if the state of termination is ok

5. If ok, then quit

6. Otherwise, apply the genetic operators for production of new population of chromosomes of membership function, and then perform step 2.

Actually we want the best set of membership function for quantitative data, that's why we use the genetic algorithm. To check the performance of the proposed method, the method is applied upon the several public data sets (Pima, iris, heart, and thyroid) and compared the results with the previous methods. The results of the proposed methods are very good in terms of accuracy, comprehensibility, number of rules generated. The FDT is small in depth wise. The only thing which we consider as a limitation is that it is time taking as compare to existing approaches.

### 2.3 Genetic Algorithm based Technique

The genetic algorithm is also used for the sake of rule extraction. The extracted rules by genetic algorithm are very few in numbers but these extracted rules are significant. The only problem in rule extraction with the genetic algorithm is its complexity. The studied technique is about the selection of fuzzy if-then rules for classification problem using GA [33]. The paper presented a method by the use of genetic algorithm as its base to select fewer and important fuzzy if-then rules in order to construct the dense fuzzy classification system which has classification power. This is a kind of optimization problem with two objectives: i.e. maximum number of patterns should be classified correctly and the number of rules should be minimized. In the regard of minimizing the number of fuzzy if-then rules and setting of their membership functions several techniques have been used but few of them
deal with the classification problem. The performance of any fuzzy classification system relies on the fuzzy if-then rules and selection of fuzzy partition being used. If the partition is too rough or uneven then it might be possible that it gives sow performance because there is chance of misclassification of patterns. If partition is good then it might be possible that the system cannot generate enough if-then rules due to deficiency of training patterns. So the selection of fuzzy partition is very important. For understanding assume the Figure 2.4 which is elaborates the two class classification problem.

In the Figure 2.4 there are two types of circles like open circles and closed circles. Open circle representing the class-1 whereas the closed circles representing the class-2. For this type of classification problem a fine fuzzy partition should be needed for the pattern space which is on the right side and uneven fuzzy partition should be beneficial for the pattern space on the right side. So we concluded that for such kind of pattern space it is difficult to define the fuzzy partition which can deal with two opposite conditions. To handle this issue the concept of distributed fuzzy if then rules [34] emerged. In it all the fuzzy if-then rules associated to different fuzzy partitions are employed at the same time in fuzzy inference. The only drawback is
that when a high dimensional space problem comes then lot of fuzzy if then rules constructed. This problem can be overcome by the elimination of unnecessary rules from the rule list and selection of relevant rules which cover maximum number of patterns. In this way a dense classification system developed with huge classification power. The advantages of this proposed system are.

- Need normal storage capacity.
- High inference speed for new patterns.
- User can examine every fuzzy if-then rule easily.

The proposed system is tested on the Iris dataset of Fisher, the results and simulations generated by the systems are better and provides well classification accuracy. The classification percentage is near about 99% for Iris dataset. The generated fuzzy if then rules minimized by the 2% of the total rules which are 692. Since the system selects the substantial classification rules so we it can be used as a knowledge acquisition tool for classification problems.

2.4 Summary

The chapter literature review is mixture of various techniques that have been discussed in detailed. The techniques have been classified into three major areas like ACO (ant colony optimization), Fuzzy logic, and GA (genetic algorithm) based techniques. Each area comprises a lot of work but we have explained here a very little and relevant work which would be helpful for us in understanding the proposed problem statement. In the section ACO based technique we have discussed the evolution of ant miner and its versions. Fundamentally the use of ACO for the very first time as rule extraction is named as ant miner, then with some changes the second version ant miner2 evolved after it ant miner3 and ant miner+ respectively with major or minor changes. All the discussed versions of ant miner are addressing the area of classification. The difference between the versions of ant miner is the computation overhead, changing of the mathematical functions, but the version ant miner+ is different from the previous versions because the environment of ants to move from source to destination is defined in terms of graph. In fact each version of ant miner is for the sake of improvement of classification rate and accuracy for the
classification system. The results with each version is promising with respect to previous version with the little bit tradeoff of time taken by the classifier, the rules extracted by the classifier, the accuracy rate and classification rate given by the classifier. In genetic algorithm area the aim is the same to extract the rules, but here the objective is the maximum number of patterns should be classified with the use of fewer if-then rules. Similarly in the area of fuzzy techniques the different fuzzification techniques has been discussed through which we can change the fuzzy linguistic function with sharp edges to smooth or curve edges which can cover the more patterns and the built system will give the more accuracy because crisp decisions replaced by the fuzzified boundaries. In the same way fuzzy if-then rules extracted which gives high accuracy as compared to the methods that have been implemented without knowing the importance of fuzzy logic and its impact upon the overall final results.
3 Problem Statement

3.1 Problem statement

There are two problem statements that have discovered and found in the base paper.

- Problem 1: Partitioning of fuzzy membership function is performed uniformly without using any Quantitative Technique.
  Fuzzy memberships functions are partitioned into the equal intervals into linguistic terms without use of any quantitate methodology. The Figure.3.1 shows the partitioning of the fuzzy membership functions.

![Figure 3.1: Partitioned fuzzy membership function showing five fuzzy sets. 1: small, 2: medium small, 3: medium, 4: medium large, 5: large.](image)

- Problem 2: The performance of FCS-Antminer (fuzzy classification system) [16] technique has not been investigated over Trapezoidal and Bell shaped fuzzy membership functions, only Triangular fuzzy membership function have been discussed as shown in figure 3.1.
  The base paper focuses on the triangular fuzzy membership function only. It means that the capacity is to deal the categorical values only. The continuous values are discretized at first and then make them use into the method. For discretization of the values a function will be defined in the method which will convert the continuous values into categorical. This increases the computation time of the method and make the method more complex. The one thing more due to which the triangular fuzzy
membership function works not well is accuracy of the method. As we know that the triangular fuzzy membership function have crisp boundaries or in other words we can say that they it has sharp edges, due to this the decisions made by this type of fuzzy sets are tight and there is more chances that the values that may lie near to the crisp edges lost and couldn’t be consider during training of entire method. It may happen that when one partition of fuzzy set over lie into the other fuzzy partition then the values that are very near to the point of their lying or cross wouldn’t be consider because of crisp decision. Furthermore while over lying of the fuzzy sets it may possible that the half of the value lie in one partition and half lie in 2\textsuperscript{nd} partition. At this stage the decision taken is very difficult by the method to make that value under consideration into the specific fuzzy partition. If we use the trapezoidal and bell shape fuzzy membership function then the accuracy rate of the classifier or the method would be increased significantly. The classification rate will also be changed by the use of trapezoidal and bell shape fuzzy membership functions. The examples of triangular fuzzy membership function and trapezoidal fuzzy membership function have shown in Figure 3.2 and in Figure 3.3 respectively.

![Figure 3.2: Triangular fuzzy function](image1)

![Figure 3.3: Trapezoidal fuzzy function](image2)

### 3.2 Research Objectives

The research objectives behind this research is to facilitate the doctor’s community and provide them an enough accurate classification system whose accuracy will be the best of all the previous versions that are so far designed for the classification of diabetes disease. In medical field there are lot of discrepancies from the point of
disease classification, the reason of it is much similarity between test results and symptoms shown in the reports of that particular test results. These discrepancies related to the early diagnose of the correct disease, because some diseases are complicated and dangerous, due to their late recognition a patient may suffer severity. Only a physician with good knowledge and with good experience can classify from the test results that from which disease the patient is actually suffering. This is all due to the fact that some disease are so much complicated that their symptoms are mixing up with the other disease, so at that stage physician have to diagnose the actual disease on his previous knowledge and practice. The wrong diagnostic may lead a patient to death because the actual disease which is still hidden and will start its worse effects on the body by spreading the gems in the whole body with the passage of time. So our work is on exploration of such type of disease known as “Diabetes Mellitus” and this particular disease is a type of diabetes disease. The diabetes disease existence is of two types: one is known as type-1 and second is known as type-2 or diabetes mellitus (on which we are working). The type-2 disease is difficult to diagnose because of similar symptoms with other diseases.
4 Proposed Solution

In chapter proposed solution the proposed methodology is discussed in detail which will ultimately leads us to acquire the desired results for the problem statements that are stated in the chapter 3. The main and difficult challenge is the way to find the method through which values of attributes can be fuzzified. Obviously there is need of fuzzy membership function which can consider and as well as perform well on each value of attribute for the given dataset. After Fuzzification of values the extraction of rules is big challenge, and this challenge cannot be fulfilled without the use of fuzzified values. So Fuzzification is important but for Fuzzification the fuzzy membership function plays the vital role. The detailed discussions and analysis of results and experiments are also the part of this chapter. At the end the chapter comprises of summary section, the summary section encloses the bird eye view of the whole chapter.

4.1 Proposed Architecture

The Figure 4.1 shows the architecture of our proposed methodology and as well as the algorithm of our proposed method in the form of figure. The architecture is in generic form in figure 4.1. The following two modules are important and considered as main, because the whole strategy of moving towards the solution is hidden in these.

1. Gini Index
2. ACO based rule generation

The architecture and working of these modules are explained in subsequent sections with each and every aspect. The proposed architecture shown in Figure 4.1 works as the algorithm begin with the start (main) function, after starting from the main function the next goal is to find the rule-set, the rule-set have the list of all the rules that have been discovered by the Ants. Initially rule-list initialized to zero because there is no rule at the start of the algorithm. After initializing the rule-set the next step is to find the Gini Index [16].
The fuzzy membership function applied on the given dataset to fuzzify the values of attributes. The detailed and step by step procedure involved in gini fuzzification of attribute values explained later in detail. When the values of all the attributes are fuzzified then the flow of algorithm goes to rule-learn function for each class. If there is $k$-number of classes then the rule learning process will learn the rule for each class. In our dataset the classes are two. The rule learning process will start in a way
that the ant colony optimization method will take under consideration the gini fuzzified values for the sake of rules extraction. Here at this stage the ant colony optimization algorithm utilized as whole to extract the rules, its working is explained in the separate section. After discovering the rules the stopping criteria will be evaluated before addition of constructed rule to the rule learn list. The stopping criteria is “when a rule constructed by the ant is followed by all the ants and there is no change” then it means that the learning process for that rule is stopped so the constructed rule is added to the discovered rules and then this rule pass to the rule-set. If somehow he stopping criteria not meets then the rule learning process will start again from the rule-learn function. This process will continue iteratively until all the rules are discovered which can cover each and every possibility available in the whole dataset so that on the basis of extracted rules the unseen patterns can be classified or predicted. After discovery of all the rules the certainty of grade will be computed for each discovered rules. The rule which have high certainty will be the best rule among all the discovered rules. After this the algorithm stops and it can now classify the patterns efficiently.

4.2 Gini Fuzzy membership function

As discussed earlier that the gini fuzzy membership functions is used to fuzzify the values of attributes for a given dataset. The Figure 4.2 showing the fuzzy membership functions in the form of flow chart. It is capable to fuzzify the values in a way that triangular fuzzy membership function can easily be represented in trapezoidal fuzzy membership functions. To compute the Gini-index fuzzification function, the designed fuzzy membership function has to apply on each value of given attribute for desire dataset to fuzzify the values. To find a fuzzy membership of a particular value of attribute the function (inside gini index [16]) fuzzy membership called.
The function passes three arguments (a, v, left) where “a” is the given attribute “v” is the split point value and argument “left” shows that whether the attribute value to whom we have to find the fuzzy membership is greater than split point value or less then the split point value. If the attribute value is less than the split point value then the fuzzy membership function will be called to fuzzify the value. The fuzzy
membership function pass two arguments (attribute a, split_point v), where “attribute a” is the value of given attribute and “split_point v” is the value of split-point. The split-point will be defined at that point where the class information of given attribute change, split-point value calculated by adding the two values of given “attribute a” and divided by the number of classes which have to classify in the data set, in our data set the classes are two so we divide it by the number 2. To calculate the Standard-deviation for the given “attribute a” is also the part of defined fuzzy membership function. The main goal starts to fuzzy the attribute value. To find the fuzzy value there is condition which checks that the attribute value for which fuzzy value have to calculate is less or greater than the split-point value, whether it is less or greater in both cases fuzzy-membership value is calculated and attribute’s value will be fuzzified on the basis of function which is expressed in the Equation.4.1.

\[
fuzzy_value = \begin{cases} 
1 - 1/\left(1 + \exp(-\sigma \cdot (\text{val} - \text{splitpoint}))\right) & \text{if } \text{val} \leq \text{splitpoint} \\
1/\left(1 + \exp(-\sigma \cdot (\text{val} - \text{splitpoint}))\right) & \text{if } \text{val} \geq \text{splitpoint} 
\end{cases}
\]  
(4.1)

### 4.3 ACO based rule generation

The mechanism of rules extraction through ant colony optimization is represented in the Figure 4.3. The first step is the creation and releasing of ants form the source to the destination or food. The pheromone management step belongs to initialization of pheromone so that the ants can follow the path by smelling the pheromone. The selection of path or term by the ant is dependent upon the pheromone level which is so called probability and selection process. When the rule is extracted by the ant then the quality of that rule will be calculated. The quality depends upon the pheromone level available on the current path and how many ants will likely to follow the current path or extracted rule. If all the ants able to follow the built path then the rule will be added to learned rules list (otherwise control goes to second step manage pheromone, pheromone initialized again) and the pheromone level for that particular path will be updated. It will lead to solution if all the ants created otherwise control go to first step creation and releasing of ants.
Figure 4.3: Rule extraction mechanism of ACO.
4.4 Gini Index

Gini index calculated for each of the split point against each attribute of the given data set. The gini index is the split measure that is used to know about the best split in fuzzy decision tree algorithms. The formula used for gini index is expressed in Equation 4.2.

\[
D(x_i) = \sum_{i=1}^{C} \frac{N^{(v)}}{N^{(a)}} \left[ 1 - \sum_{k=1}^{C} \left( \frac{N^{(v)}_{w_k}}{N^{(v)}} \right)^2 \right]
\]  

(4.2)

Where

C represents the total number of classes in the given data set.

V belongs to the total number of partitions.

\(N^{(a)}\) represents the sum of fuzzy membership values for all the records of an attribute of the given data set, before split.

\(N^{(v)}\) used for the sum of fuzzy membership values for all the instances in \(v_{ih}\) partition.

\(N^{(v)}_{w_k}\) is the sum of the product of fuzzy-membership values of the attribute and the fuzzy-membership values of the corresponding instances for the class \(w_k\) in the \(v_{ih}\) partition.

We can apply the gini index formula on the values that are fuzzified by the fuzzy membership function. The process of Fuzzification and the fuzzy membership function is already explained in the form of flow char and as well as equation. After applying the gini index formula and calculating its value against each split, we will come to know that which attribute has the ability to select as the root node. The attribute which have the highest gini index value against the split point will be consider as the root node of the tree. The process continues iteratively for all the
attributes against their split points till the given attributes sorted and forms a tree like structure.

4.5 **Experimental results and Discussions**

4.5.1 **Date set**

For our experiments we will use Pima Indian Diabetes dataset (PID), and this data set is used by many data mining approached for the sake of classification. The data set is taken from the UCI machine learning respiratory Blake and Merz [22]. The data set (PID) contains 768 tuples for female patients of Pima Indian heritage, it contains 8 attributes which are integer-valued and there is one class variable which has two values but for individual tuple it can have only one. A brief description of Data set has been shown in Table 4.1.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>No. Of Samples</th>
<th>No. of Classes</th>
<th>No. of Attributes</th>
<th>Sample distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>768</td>
<td>2</td>
<td>8</td>
<td>Class1: normal (500)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Class2: pima Indian diabetes (268)</td>
</tr>
</tbody>
</table>

Table 4.1: Brief description of Data set

A subset of entire data set has been shown in Table 4.2.

<table>
<thead>
<tr>
<th>No. of times pregnant</th>
<th>Plasma glucose concentration</th>
<th>Diastolic blood pressure</th>
<th>Triceps skin fold thickness</th>
<th>Serum insulin</th>
<th>Body mass index</th>
<th>Diabetes pedigree</th>
<th>Age (years)</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>148</td>
<td>72</td>
<td>35</td>
<td>0</td>
<td>33.6</td>
<td>0.627</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>85</td>
<td>66</td>
<td>29</td>
<td>0</td>
<td>26.6</td>
<td>0.351</td>
<td>31</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>183</td>
<td>64</td>
<td>0</td>
<td>0</td>
<td>23.3</td>
<td>0.672</td>
<td>32</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>89</td>
<td>66</td>
<td>23</td>
<td>94</td>
<td>28.1</td>
<td>0.167</td>
<td>21</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>137</td>
<td>40</td>
<td>35</td>
<td>168</td>
<td>43.1</td>
<td>2.288</td>
<td>33</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>116</td>
<td>74</td>
<td>0</td>
<td>0</td>
<td>25.6</td>
<td>0.201</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>78</td>
<td>50</td>
<td>32</td>
<td>88</td>
<td>31.0</td>
<td>0.248</td>
<td>26</td>
<td>1</td>
</tr>
</tbody>
</table>
4.5.2 Performance Measures

The performance of proposed system is evaluated by three different criteria.

4.5.2.1 Classification Rate

The classification rate is the number of samples correctly classified and being calculated according to Equation 4.3

\[
Classification\ rate = \frac{(TP + TN)}{TP + TN + FN + FP} \tag{4.3}
\]

Where

TP represents true positive, the number of cases in our training set covered by the rule that have the class predicted by the rule.

FP represents false positives, the number of cases covered by the rule that have a class different from the class predicted by the rule.

FN represents false negatives, the number of cases that are not covered by the rule but that have the class predicted by the rule.

TN represents true negatives, the number of cases that are not covered by the rule and that do not have the class predicted by the rule.
4.5.2.2 Precision and Recall

Precision is the number of correctly classified samples which are predicted as diabetic samples whereas the recall refers the number of classified samples which are actually diabetic by nature. Both precision and recall calculated by the Equations 4.4 and 4.5 respectively.

Precision and recall have inverse behavior to each other [31]. As precision increases, then recall decreases and when recall increases then precision decreases.

\[
\text{Precision} = \frac{TP}{TP + FN} \quad (4.4)
\]

\[
\text{Recall} = \frac{TP}{TP + FP} \quad (4.5)
\]

4.5.2.3 F-Measure

F-Measure used as trade-off among the precision and recall. It is harmonic-mean of precision and recall and takes account of both measures. It is calculated according to the Equation 4.6.

\[
F - \text{Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4.6)
\]

4.5.3 Simulation Tool

Matlab is used as simulation tool and for implementation of our proposed methodology.
4.5.4 Results

Pima Indian Diabetes dataset have been taken from the UCI Machine learning repository. On this dataset different types of experiments carried out and results are generated. The results are mixture of numerical and graphical in the form of figures.

4.5.4.1 Numerical Results

The results of proposed methodology are compared with the results of other state of art methods that have been evolved and built to classify the patterns specific to dataset Pima Indian Diabetes disease. The Table 4.3 depicts the classification accuracy of various state of art methods that have been studied and proposed by different researchers. Some of the mentioned methods are implemented on Weeka tool and some of them are implemented in any language or tool. From the table it is clear that the classification accuracy range of mentioned methods is above then the 70 and below then the 85 percent. So we can say that the minimum classification accuracy achieved is 70.99% and the maximum classification accuracy is 84.24% which is the best till now. This classification accuracy varies according to method and technique being used. Each tried its best to achieve the best classification accuracy. If we analyze according to the versions of ant miner then we will come to know that at the beginning ant miner could not produce good results with respect to the other techniques that already gave the good results but with the passage of time as more versions of ant miner came the classification accuracy increased and reached to the 84.24%, which is the best amongst all. The analysis according to ant miner is necessary because ant miner is the state of art method in which ant colony optimization algorithm was used first time for rules extraction. Then the later versions used the same concept of rules extraction but with different computation technique and mathematical functions. Our proposed technique is also based upon the ant colony optimization method for rule generation but with some addition like gini fuzzification technique which helps to improve the overall result and accuracy because the fuzzy membership function with crisp decisions have been omitted by using the fuzzified values based fuzzy membership function. Our proposed
methodology proved this by implementing this concept and able to get the classification accuracy to 87.12%, which is the best accuracy amongst all the other state of art methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification Accuracy (%)</th>
<th>Study</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLNN with LM (10*FC)</td>
<td>76.62</td>
<td>Temurtas et al (2009)</td>
</tr>
<tr>
<td>MLNN with LM</td>
<td>82.37</td>
<td>Temurtas et al (2009)</td>
</tr>
<tr>
<td>Smart</td>
<td>76.8</td>
<td>Polat et al (2008)</td>
</tr>
<tr>
<td>Log disc</td>
<td>77.7</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>QDA</td>
<td>59.5</td>
<td>Polat et al (2008)</td>
</tr>
<tr>
<td>CART</td>
<td>72.8</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>Regression coefficients</td>
<td>72.3958</td>
<td>Witten &amp; Frank (2005)</td>
</tr>
<tr>
<td>ASR</td>
<td>74.3</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>LFC</td>
<td>75.8</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>kNN</td>
<td>71.9</td>
<td>Polat et al (2008)</td>
</tr>
<tr>
<td>ID3</td>
<td>71.7</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>Kohonen</td>
<td>72.7</td>
<td>Kahramanli&amp;Allahverdi (2008)</td>
</tr>
<tr>
<td>RBF</td>
<td>75</td>
<td>Witten &amp; Frank (2005)</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>75.30</td>
<td>Witten &amp; Frank (2005)</td>
</tr>
<tr>
<td>SVM</td>
<td>76.0</td>
<td>Witten &amp; Frank (2005)</td>
</tr>
<tr>
<td>Bayesian logistic regression</td>
<td>72.39</td>
<td>Witten &amp; Frank (2005)</td>
</tr>
<tr>
<td>GRNN</td>
<td>80.21</td>
<td>Kayaer &amp; Yildirim (2003)</td>
</tr>
<tr>
<td>SA</td>
<td>75.71</td>
<td>Mohammadi et al (2008)</td>
</tr>
<tr>
<td>Antminer</td>
<td>70.99</td>
<td>Mohammadi et al (2007)</td>
</tr>
<tr>
<td>Antminer2</td>
<td>76.58</td>
<td>Mohammadi et al (2007)</td>
</tr>
<tr>
<td>FCS-Antminer</td>
<td>84.24</td>
<td>Mohammadi et al (2011)</td>
</tr>
<tr>
<td>Gini Antminer</td>
<td>87.13</td>
<td>Our study</td>
</tr>
</tbody>
</table>

Table 4.3: Classification accuracy obtained by different methods for PID.
The Table 4.4 is the brief explanation and comparison of the classification rate of proposed method with other state of art methods that have been evolved for the purpose of classification. The mentioned classification rate is in percentage. The classification rate of mentioned state of art methods can be comparable with each other. Each have different classification rate with respect to other because of the tool and techniques used for each particular method is different. The important thing of the table is that the mentioned classification rate for each method is obtained on PID (Pima Indian diabetes) dataset. The classification rate can be analyzed easily. The classification rate range is from 67 to 87.21%. The minimum classification rate is 67% whereas the maximum classification rate is 87.21%. Our proposed technique comes up with highest classification rate amongst the all other mentioned methods. This is our aim and our goal which we have achieved by the use of ant colony optimization method for rules extraction and gini fuzzification function to fuzzify the values, so that crisp decisions eliminated permanently.

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4.5 rules</td>
<td>67.0</td>
</tr>
<tr>
<td>SMART</td>
<td>76.8</td>
</tr>
<tr>
<td>Decision table</td>
<td>71.22</td>
</tr>
<tr>
<td>Bayes</td>
<td>72.2</td>
</tr>
<tr>
<td>Regression coefficient</td>
<td>72.3</td>
</tr>
<tr>
<td>C4.5</td>
<td>75.4</td>
</tr>
<tr>
<td>NNGE</td>
<td>73.56</td>
</tr>
<tr>
<td>RBF</td>
<td>75.8</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>75.3</td>
</tr>
<tr>
<td>SVM</td>
<td>77.3</td>
</tr>
<tr>
<td>LS-SVM</td>
<td>78.21</td>
</tr>
<tr>
<td>SA</td>
<td>75.71</td>
</tr>
<tr>
<td>FCS-Antminer</td>
<td>84.32</td>
</tr>
<tr>
<td>Gini Antminer</td>
<td>87.21</td>
</tr>
</tbody>
</table>
4.5.4.2 Graphical Results

The Figure 4.4 shows the effect of number of ants on the classification rate. The Figure 4.3 is showing two lines. The top line graph is showing the result of proposed technique whereas the below line graph is showing the result of base paper technique (FCS Antminer). The difference of results is clear, the proposed technique performing well from the start, there are some points in the middle where the difference between the results are not so much, but as the number of ants increasing the classification rate is also increasing and it keeps on increasing till the number of ants reached to 200. From 200 ants to 500 ants the classification rate becomes stable and there is no more increase in that, so we can say that the classification rate achieved by the proposed technique is 87.21 for 200 to 500 number of applied ants, whereas if we view the result of Fcs Antminer technique then we will see that the maximum classification accuracy achieved was 84.21 for 200 to 500 number of ants. It also states that the proposed technique results are better for the minimum to maximum number of ants. It also noticeable that in both cases the classification is minimal for low number of ants but as the ants increases then the classification rate is also increasing gradually. It’s because of the fact that from the large number of ants the classifier or the system will be able to extract the rule which will be the refined one and covering the patterns with every aspect.
The Figure 4.5 shows the effect of number of ants to build the classifier with respect to time domain and the time calculated is in seconds. The Figure 4.5 is showing the result of proposed technique and as well as the result of FCS Antminer technique. The figure is showing that when the ants are fewer in number then time required by the classifier is minimal and there is not too much difference in the result of two different techniques. When the number of applied ants is 700 then there is no much difference in the result but as the applied number of ants’ increases from 700 to 1000 then the result is clear that the proposed technique took much more time with respect to the FCS Antminer technique. The reason of increase in time for the proposed technique is the mathematical computations involved in it, computations like gini fuzzification to fuzzify the values of each attribute. The word classifier means the classification system building which is used to solve the classification problems.
The Number of Terms parameter is vital element in this method. The change in this parameter brings the change in overall performance of the method. Basically the rule which is extracted by the ant colony optimization method is the combination of terms. The term itself is combination of triplets like (attribute, operator, value) such as (Sex=Male). So each extracted rule has the terms. So terms are very important because without terms a rule cannot be made. The term itself is the name of attribute, operator and the value of attribute. One rule can be composed of eight terms, because in our dataset the number of attributes is eight. So to predict the pattern or instance we have to use its possible values so that we can extract the best rule.

The Figure 4.6 shows the effect of the Number of Terms on the Classification rate. The maximum classification rate obtained when the value of parameter i.e. Number of Terms is 3. The value 3 for parameter i.e. Number of Terms means that the rule comprises of 3 terms or a new ant can alternate the discovered rule by changing the
values of 3 terms at most. The classification rate keeps on decreasing if the altered terms in the discovered rule are more than 3 or if the discovered rules have more than 3 terms as whole. This is because the artificial ants are intelligent and they have sharp memory, so if we use more and more terms in the rule then the ants will save these terms in their memory and must apply those on each visit. As a result the constructed rule will become complex in modification due to implication of many terms in the rule. It will degrade the classification rate. On the other hand if we restrict the parameter Number of Terms to 3 which means that an ant have to discover the comprehend rule by using the combination of 3 best terms. If a rule have more than three terms than the new ant can alter 3 terms at the most in order to modify the rule. In this way our proposed method will give the best classification rate as shown in the Figure 4.6.

Figure 4.6: Effect of Max_Change parameter on classification rate.
4.6 Summary

In this chapter simulations have been performed on the Pima Indian Diabetes dataset by fuzzifying the values in a way that the values can lie in the fuzzy trapezoidal membership function, and as well as the rules are extracted on the basis of fuzzified values by using ant colony optimization method. The results are evaluated by considering the classification rate and classification accuracy with different state of art methods. Simulation results showed that the proposed solution performed well and have given good classification rate and accuracy with respect to other state of art methods. By fuzzifying the crisp edges the performance of classifier increases incredibly.
5 Conclusions and Future Work

5.1 Conclusion

As we have seen in literature review that ACO algorithm used for classification by extracting the rules. The four versions of ant miner proposed and each version is based upon the ACO algorithm. Each new version have some different approached in the technique, in the function and change in the mathematical computing formulas. In our technique the new thing is that we have used the Gini index with the ACO to get better results and more accuracy. We have proved this by applying it on the Diabetes dataset. The results of our techniques are promising and better than the results generated by all the previous discovered techniques. By using trapezoidal fuzzy membership functions instead of triangular fuzzy membership function, the classification rate and accuracy increased. Automatic computation of fuzzy membership function is more robust than constructing these functions based on data distribution.

5.2 Limitation

The limitation of our proposed method is its computation overhead. There is lot of computation involved in it which is time taking process. Computation like calculation of Gini index for each attribute, application of fuzzy membership function on each value of attribute and extraction of rules through ant colony optimization algorithm.

5.3 Future Work

In future more improvements can be brought up by optimizing the fuzzy membership functions using the optimization techniques. The performance of classification can be enhanced by using the more fuzzy membership function with some optimization algorithm. Moreover the computation time can be decreased by using any method other than the gini index fuzzification.
6 References


